Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for managing digital data, ~~said method~~ comprising the steps of:

associating digital data with a first predefined set of digital data, whereby at least two predefined sets of digital data exist and said first predefined set of digital data can be distinguished from the other predefined sets of said at least two predefined sets;

computing a first leaf hash value over some or all of the digital data and/or over identifications of some or all of the digital data that are associated with said first predefined set;

computing a second leaf hash value over some or all of the digital data and/or over identifications of some or all of the digital data that are associated with a second predefined set of said at least two predefined sets;

if more than two predefined sets exist, for each remaining set of said at least two predefined sets~~:~~

computing respectively a leaf hash value over some or all of the digital data and/or over identifications of some or all of the digital data associated with a remaining predefined set; and

computing a root hash value, whereby the underlying hash algorithm has as an input at least said leaf hash values that are respectively computed for each of said at least two predefined sets of digital data, whereby said step of computing a root hash value comprises ~~a step of:~~

computing a first non-leaf hash value over at least said first and said second leaf hash value;

said method further comprising steps for determining the consistency of given digital data with said root hash value comprising:

identifying the set of digital data that is associated with given digital data;

re-obtaining said root hash value;

re-obtaining the hash values over which said root hash value was computed, comprising a step of:

re-computing the leaf hash value over some or all of the digital data and/or over identifications of some or all of the digital data associated with said identified set of digital data applying the same computing scheme as in said steps of computing a first and a second leaf hash value;

computing a hash value over said re-obtained hash values applying the same computing scheme as in said step of computing a root hash value;

comparing said re-obtained root hash values with said in the previous step computed hash value; and

determining the consistency of said given digital data with said root hash value based upon said comparing step, whereby consistency is determined if said comparing step results in equal hash values.

2. (Currently Amended) The method according to claim 1, wherein at least one of the following identifications is assigned to or obtained from each of said digital data:

a number associated with the digital data;

a unique identification of the digital data;

a predefined part of a unique identification of the digital data;

a predetermined number of bits extracted from predetermined bit positions of a digital identification associated with the digital data;

a predefined search string associated with or comprised in the digital data;

a time stamp of the digital data;

a hash value of the whole digital data;

an identifier especially created for this purpose and added to said digital data; and

a hash value of at least one of the above identifications.

3.   (Currently Amended) The method according to claim 2, wherein said step of associating digital data to a first predefined set of digital data is accomplished using said at least one identification.

4.   (Currently Amended) The method according to claim 2 or 3, wherein

said at least two predefined sets are identified and/or distinguished by said identifications; and wherein

said step of identifying the set of digital data that is associated with given digital data comprisingcomprises:

determining the at least one identification of said given digital data, and whereby said identifying is accomplished using said at least one determined identification.

5.   (Currently Amended) The method according to any one of claims 2 to 4claim 2, wherein said step of computing a first leaf hash value, said step of computing a second leaf hash value and/or said step of computing respectively a leaf hash value for each remaining predefined set if more than two predefined sets exist comprising:

obtaining the at least one identification for some or all of the digital data associated with said predefined set; and

computing a hash value over said in the previous step obtained identifications; and wherein

said step of re-computing the leaf hash value for said identified set of digital data comprisingcomprises:

re-obtaining the at least one identification for some or all of the digital data associated with said identified set; and

computing a hash value over said re-obtained identifications.

4

6. (Currently Amended) The method according to ~~any one of claims 2 to 4~~claim 2, ~~wherein~~said step of computing a first leaf hash value, said step of computing a second leaf hash value and/or said step of computing respectively a leaf hash value for each remaining predefined set if more than two predefined sets exist comprising:

for some or each digital data associated with said predefined set computing respectively a hash value of the at least one identification;~~,~~ and

computing a hash value over at least said in the previous step computed hash values; ~~and wherein~~

wherein said step of re-computing the leaf hash value for said identified set of digital data ~~comprising~~comprises:

obtaining the at least one identification for some or each of the digital data associated with said identified set;~~,~~

computing respectively for each in the previous step obtained identification a hash value;~~,~~ and

computing a hash value over at least said in the previous step computed hash values.

7. (Currently Amended) The method according to ~~any one of claims 1 to 4~~claim 1, ~~wherein said method~~ further comprising:

for each of said at least two sets of digital data computing respectively a hash value, comprising computing of each of the digital data associated with said set respectively a hash value; ~~and wherein~~

wherein said step of computing a first leaf hash value, said step of computing a second leaf hash value and/or said step of computing respectively a leaf hash value for each remaining predefined set if more than two predefined sets exist ~~comprising~~comprises:

computing a hash value over at least said in the previous step computed hash values of the respective set of digital data; and ~~wherein~~

said step of re-computing the leaf hash value for said identified set of digital data comprising:

re-computing respectively a hash value of each of the digital data associated with said identified set;~~,~~ and

computing a hash value of said respectively re-computed hash values.

8.   (Currently Amended) The method according to claim 7, wherein said step of computing respectively a hash value for each of said at least two sets of digital data is comprised in said steps of computing a first leaf hash value and/or said step of computing a leaf second hash value and/or said step of computing respectively a leaf hash value for each remaining predefined set if more than two predefined sets exist.

9.   (Currently Amended) The method according to ~~any one of claims 1 to 8~~claim 1, wherein at least four predefined sets of digital data exist that can be distinguished from one another, ~~said method~~and further comprising:

computing a third leaf hash value over some or all of the digital data and/or over identifications of some or all of the digital data associated with a third predefined set of said at least four predefined sets;

computing a fourth leaf hash value over some or all of the digital data and/or over identifications of some or all of the digital data associated with a fourth predefined set of said at least four predefined sets; and wherein

said step of computing a root hash value further ~~comprising~~comprises:

computing a second non-leaf hash value over at least said third and said fourth computed leaf hash value~~;~~; and

computing a third non-leaf hash value over at least said first and said second computed non-leaf hash value.

10.   (Currently Amended) The method according to claim 9, wherein said computed leaf hash values, non-leaf hash values and root hash value represent a tree structure, and wherein:

said first, said second, said third and said fourth leaf hash value are associated with a bottom layer of said tree structure;

said first and said second non-leaf hash value are associated with a second layer of said tree structure;

said third non-leaf hash value is associated with a third layer of said tree structure; and

said root hash value is associated with a top layer of said tree structure;,

wherein said top layer is the highest layer of said tree structure comprising only a single non-leaf hash value, which is said root hash value.

11. (Currently Amended) The method according to claim 10, wherein further predefined sets of digital data exist, each associated with a further leaf hash values computed in the same manner as said first to fourth leaf hash value, and/or

further layers of said tree structure exist, each layer being associated with further non-leaf hash values computed over some or all of the non-leaf hash values of a respectively lower layer in said tree structure.

12. (Currently Amended) The method according to claim 10 or 11, wherein said step of re-obtaining the hash values over which said root hash value was computed further comprising:

re-computing each non-leaf hash value in the second layer of said tree structure that was computed over hash values comprising the re-computed leaf hash value of said identified set of digital data; and

for each of the third to the top layer of said tree structure, re-computing each non-leaf hash value that was computed over hash values comprising a non-leaf hash value that was re-computed in this step or said previous step.

13. (Currently Amended) The method according to claim 12, wherein said step of re-obtaining the hash values over which said root hash value was computed further comprising:

providing all remaining leaf hash value required in said step of re-computing each non-leaf hash value in the second layer of said tree structure that was computed over hash values comprising the re-computed leaf hash value of said identified set of digital data; and

providing all remaining non-leaf hash value required in said step of re-computing each non-leaf hash value for each of the third to the top layer of said tree structure.

14.   (Currently Amended) The method according to ~~any one of claims 10 to 13~~claim 10, ~~wherein~~ said step of computing the root hash value further comprising:

dividing said predefined sets into a plurality of groups of predefined sets;

computing for each of said groups of predefined sets a non-leaf hash value over the computed leaf hash values of the predefined sets in said group; and

for each of the third to the top layer of said tree structure dividing the non-leaf hash values of the next lower layer into at least two groups and computing a non-leaf hash value over the non-leaf hash values of each group.

15.   (Currently Amended) The method according to claim 14, wherein for each layer L of said tree structure said predefined sets and/or said non-leaf hash values of the next lower layer $L-1$ are divided into groups of a predefined number B_L of predefined sets or non-leaf hash values.

16.   (Currently Amended) The method according to claim 15, further comprising:

determining whether the total number of predefined sets is an integer multiple of said predefined number B_1;

if said total number is not an integer multiple of B_1 creating at least one group comprising respectively less than B_1 predefined sets, or creating at least two groups both comprising at least one identical predefined set;

determining for each of said second to said top layer of said tree structure L whether the total number of non-leaf hash values of the next lower layer $L-1$ is an integer multiple of said predefined number B_L; and

if said total number is not an integer multiple of B_L performing one of the following:

creating at least one group comprising respectively less than B_L non-leaf hash values of the next lower layer L-1;~

creating at least two groups both comprising at least one identical non-leaf hash values of the next lower layer L-1;~ or

creating at least one group comprising at least one hash value of a lower layer L-N, wherein N>1 and L>2.

17.   (Currently Amended) The method according to ~any one of claims 1 to 16~claim 1, wherein each digital data is associated with one of said plurality of predefined sets.

18.   (Currently Amended) The method according to claims 17, wherein different sorts of digital data are associated with different predefined sets~, wherein said different sorts of digital data~.

19.   (Currently Amended) The method according to ~any one of claims 1 to 16~claim 1, wherein digital data can be associated with more than one of said predefined sets of digital data.

20.   (Currently Amended) The method according to claim 19, wherein if said given digital data is associated with more than one predefined sets of digital data, said step of determining the consistency of said given digital data with said root hash value comprises:

identifying respectively more than one predefined sets; and

re-computing a leaf hash for each of said more than one identified sets.

21.   (Currently Amended) The method according to ~any one of claims 1 to 20~claim 1, wherein said non-leaf hash values are computed over exclusive groups of leaf hash values and/or non-leaf hash values.

Application No. 10/580,092

22.   (Currently Amended) The method according to ~~any one of claims 1 to 13~~claim 1, wherein the total number of predefined sets is S, whereby S is an integer power E of an integer B.

23.   (Currently Amended) The method according to claim 22, wherein for each of said S predefined sets a leaf hash value is computed, and

said step of computing the root hash value ~~comprising~~comprises:

(a)   dividing said S predefined sets into E+1 mutually exclusive groups of B predefined sets;

(b)   for each of said E+1 groups computing a non-leaf hash value over the computed leaf hash values of the B predefined sets of said group;

(c)   dividing said in the previous step computed non-leaf hash values into exclusive groups of B non-leaf hash values;

(d)   computing for each of said exclusive groups in the previous step a non-leaf hash value over the B non-leaf hash values of said group; and

(e)   repeating steps (c) and (d) until a single non-leaf hash value is computed, which is said root hash value.

24.   (Currently Amended) The method according to claim 22 ~~or 23~~, wherein said integer B is an integer power of 2.

25.   (Currently Amended) The method according to ~~any one of claims 1 to 24~~claim 1, wherein said given digital data is distributed to a client in a server-client system comprising at least one server and a plurality of clients,

said steps of computing the leaf hash values for each predefined set of digital data, the steps of computing the non-leaf hash values and said step of computing the root hash value are ~~preformed~~performed by said server;

said steps for determining the consistency of said given digital data with said root hash value are performed by said client;

said method further comprising:

10

distributing said root hash to a plurality of clients of said server-client system.

26.   (Currently Amended) The method according to claim 25, wherein said step of distributing said root hash to a plurality of clients is performed by said server.

27.   (Currently Amended) The method according to claim 25 ~~or 26~~, wherein said root hash value is distributed among said plurality of clients by sending said root hash value by a first client to a second client, whereby said first client had previously received ~~sad~~ said root hash value from said server and/or from one or more of said plurality of clients.

28.   (Currently Amended) The method according to ~~any one of claims 25 to 27~~claim 25, wherein said distributed root hash value is associated with time-stamp or validity information.

29.   (Currently Amended) The method according to claim 28, wherein said root hash value is replaced, updated and/or requested from said server by a client based on said time-stamp or validity information.

30.   (Currently Amended) A computer-readable medium comprising instructions for controlling a client device in a server-client system, said instructions causing said client device to perform the steps of:

receiving a root hash value, said root has value being computed according to the method in any one of claims 1 to 29;

determining the consistency of given digital data with said root hash value comprising:

identifying a set of digital data that is associated with given digital data;

obtaining the hash values over which said root hash value was computed, comprising:

11

requesting some or all of the digital data and/or identifications of some or all of the digital data associated with said identified set of digital data from a server of said client-server system;

computing a leaf hash value over said requested some or all of the digital data and/or identifications of some or all of the digital data associated with said identified set of digital data;

determining the remaining leaf and non-leaf hash values that are required to compute said root hash value; and

requesting said remaining leaf and non-leaf hash values from said server;

computing a hash value over said obtained hash values applying the same computing scheme as in said step of computing a root hash value;

comparing said received root hash values with said in the previous step computed hash value; and

determining the consistency of said given digital data with said received root hash value based upon said comparing step, whereby consistency is determined if said comparing step results in equal hash values.

31. (Currently Amended) A computer-readable medium comprising instructions for controlling a server in a server-client system, said instructions causing said server to perform the following steps according to the method in any one of claims 1 to 29:

associating digital data with a first predefined set of digital data, whereby at least two predefined sets of digital data exist and said first predefined set of digital data can be distinguished from the other predefined sets of said at least two predefined sets;

computing a first leaf hash value over some or all of the digital data and/or over identifications of some or all of the digital data that are associated with said first predefined set;

computing a second leaf hash value over some or all of the digital data and/or over identifications of some or all of the digital data that are associated with a second predefined set of said at least two predefined sets;

if more than two predefined sets exist, for each remaining set of said at least two predefined sets:

computing respectively a leaf hash value over some or all of the digital data and/or over identifications of some or all of the digital data associated with a remaining predefined set;

computing a root hash value, whereby the underlying hash algorithm has as an input at least said leaf hash values that are respectively computed for each of said at least two predefined sets of digital data, whereby said step of computing a root hash value comprises a step of:

computing a first non-leaf hash value over at least said first and said second leaf hash value; and

distributing said root hash value to a plurality of clients of said client-server system.

32.   (Currently Amended) A method for providing trust levels of signatures in a system comprising a plurality of parties connected via a public network, said system providing a public key signature scheme for said pluralities of parties, said method comprising:

signing a public key PK1 of a first party by a second party using a private key SK2;

signing digital data D by said first party using the private key SK1 corresponding to said signed public key PK1;

obtaining said signed digital data D and said signed public key PK1 by a third party, whereby said first party is unknown and/or not trustworthy to said third party;

determining said second party as a signing party of said signed public key PK1;

determining whether said second party as a signing party is known and/or trusted by said third party; and

if said second party as said signing party is known and/or trusted by said third party, performing the steps of:

(a)   obtaining the public key PK2 of said second party corresponding to said private key SK2;

(b)   verifying said signed public key PK1 using said public key PK2 of said known and/or trusted signing party; and,

(c)   if the verification of said signed public key PK1 was successful, verifying the signed digital data D using said signed public key PK1.

33.   (Currently Amended) The method according to claim 32, wherein a public key is accepted for verifying a signature on digital data and/or on another public key by a verifying party only if the corresponding signing party in possession of the corresponding private key to said public key is known and/or trusted by said verifying party.

34.   (Currently Amended) The method according to claim 32 or 33, further comprising if said second party as said signing party is known and/or trusted by said third party and if the verification of said signed public key PK1 was successful,

said third party registering said first party and/or said public key PK1 as trusted for further succeeding signatures issued by said first party using said private key SK1.

35.   (Currently Amended) The method according to claim 32 or 33, wherein said digital data D comprises a public key of a further party of said system.

36.   (Currently Amended) The method according to any one of claims 32 to 35claim 32, further comprising:

if said second signing party is not known and/or not trusted by said third party, performing the steps of:

(i)   determining whether a further signing party has issued a signature of said signed public key PK1;

(ii)   if a further signing party signature of said signed public key PK1 is determined and said further signing party is known and/or trusted by said third party, performing the steps (a), (b) and (c) with the public key of said further signing party; and,

(iii)   if no further signing party of said signed public key PK1 is determined, and/or no trusted further signing party is determined, performing steps of:

1. obtaining the public key PK2 of said second party corresponding to said private key SK2;~

2. determining whether a fourth signing party has issued a signature of said second public key PK2; and~

3. if a fourth signing party has issued a signature on said second public key PK2, determining whether said fourth signing party is known and/or trusted by said third party, and if said fourth signing party is known and/or trusted by said third party, performing the steps (a), (b) and (c) with the public key PK4 of said fourth signing party.

37. (Currently Amended) The method according to claim 36, wherein the public key PK4 of said fourth signing party is signed by a fifth signing party, said method further comprising:

4. if said fourth signing party is not known and/or not trusted by said third party, performing the steps (i) and (ii) applied to said public key PK2, and performing the step (iii) respectively applied to the public key PK4, the fifth party, and the public key PK5.

38. (Currently Amended) The method according to ~~any one of claims 32 to 37~~claim 32, wherein a public key is accepted for verifying a signature on digital data and/or on another public key by a verifying party, only if the corresponding signing party in possession of the corresponding private key to said public key is known and/or trusted by said verifying party, and wherein

an unknown public key is trusted, if said public key is signed and the signature can be verified with another trusted and/or known public key.

39. (Currently Amended) The method according to claim 38, further comprising:

establishing a trust level of a public key, whereby said trust level is based on the number of unknown public keys in the chain of public keys to a known public key that are required to accept said public key for verifying a signature.

40. (Currently Amended) The method according to claim 38 ~~or 39~~, wherein one or more public keys in said system and/or signatures corresponding to said public keys are identified to a verifying party as not trusted, and wherein said one or more public keys and/or said signatures are not used in said chain of public keys to establish whether a unknown public key is trusted.

41. (Currently Amended) The method according to ~~any one of claims 39 to 40~~claim 39, wherein a trust information is assigned to a signature of a public key or to the public key used to verify said signature, and said trust information assigned to one or more public keys in said chain of public keys is used to establish said trust level.

42. (Currently Amended) The method according to ~~any one of claims 38 to 40~~claim 38, wherein more than one chain of public keys are used to establishing a trust level of a public key, whereby the trust levels of all chains are combined to a single trust level, or the highest trust level is used.

43. (Currently Amended) The method according to ~~any one of claims 32 to 42~~claim 32, wherein said public keys and/or the signatures of said public keys are stored on a server of said system, whereby said server is connected to said parties via said public network and said public keys and/or said signatures of said public keys are distributed to a party of said system by said server and/or by another party of said system.

44. (Currently Amended) The method according to claims 42 ~~or 43~~, wherein said server aggregates all signatures that are associated with said more than one chains to establishing a trust level, and said server assigns the aggregated or combined trust level to a further signature that is issued by said server.

45. (Currently Amended) A method for providing integrity and consistency information of digital data for at least two parties of a system comprising a plurality of parties connected via a public network, said method comprising:

creating a list of identifications of digital data by a first party of said system;

computing a hash value over some or all of the identifications of said list;

associating said hash value with said list;

providing said list and said hash value to a second party of said system;

comparing one or more of identifications in corresponding list in possession of said second party with the corresponding one or more of identifications in said obtained list; and,

verifying the consistency of both lists comprising the steps of:

computing a hash value over some or all of the identifications of said obtained list;

computing or obtaining a hash value over some or all of the identifications of said corresponding list;

comparing both hash values; and,

if said comparing step results in equal hash value, establishing that both list are consistent.

46. (Currently Amended) The method according to claim 45, wherein said step of associating said hash value with said list comprises attaching said hash value to said list.

47. (Currently Amended) The method according to claim 45 or 46, wherein said step of verifying the consistency of both lists further comprises a step of:

if said comparing step results in different hash value, informing said first and/or said second party.

48.   (Currently Amended) The method according to ~~any one of claims 45 to 47~~claim 45, wherein said step of verifying the consistency of both lists further comprises a step of:

if said comparing step results in different hash value, creating an alert accessible by other parties of said system.

49.   (Currently Amended) The method according to ~~any one of claims 45 to 48~~claim 45, wherein said digital data are public key of parties in said system.

50.   (Currently Amended) The method according to ~~any one of claims 45 to 49~~claim 45, wherein said list and/or identifications in said list are digitally signed by said first party.

51.   (Currently Amended) The method according to ~~any one of claims 45 to 50~~claim 45, wherein said identification of digital data is said digital data.

52.   (Currently Amended) The method according to ~~any one of claims 45 to 50~~claim 45, wherein a group of clients maintains mutually consistent list by interchanging said list and any update of said list between all clients of said group, whereby said list comprises a set of digital data and/or identifies of digital data.

53.   (Currently Amended) The method according to claim 52, wherein at least one of the following identifications is assigned to or obtained from each of said digital data:

a number associated with the digital data~~;~~
a unique identification of the digital data~~;~~
a predefined part of a unique identification of the digital data~~;~~
a predetermined number of bits extracted from predetermined bit positions of a digital identification associated with the digital data~~;~~
a predefined search string associated with or comprised in the digital data~~;~~
a time stamp of the digital data;

18

a hash value of the whole digital data;

an identifier especially created for this purpose and added to said digital data; and

a hash value of at least one of the above identifications.

54.   (Currently Amended) The method according to claim 53, wherein said set of digital data is determined by said identifications by associating digital data to said set using said at least one identification for each digital data, or said set of digital data is determined by a client identification.

55.   (Currently Amended) The method according to claim 53 ~~or 54~~, wherein said at least two predefined sets are identified and/or distinguished by said identifications; and said method further comprising:

identifying the set of digital data that is associated with given digital data comprising determining the at least one identification of said given digital data, and whereby said identifying is accomplished using said at least one determined identification.

56.   (Currently Amended) The method according to ~~any one of claims 53 to 55~~claim 53, wherein a predetermined group of clients maintains a mutually consistent predetermined list of digital data.

57.   (Currently Amended) The method according to claim 56, wherein said predetermined list of digital data is maintained and interchanged <u>in a server-client system comprising at least one server and a plurality of clients, said steps of computing the leaf hash values for each predefined set of digital data, the steps of computing the non-leaf hash values and said step of computing the root hash value are performed by said server;</u>

<u>said steps for determining the consistency of said given digital data with said root hash value are performed by said client; said method further comprising:</u>

<u>distributing said root hash to a plurality of clients of said server-client system.</u>
~~according to said method of any one of claims 25 to 29.~~

19

58.   (Currently Amended) The method according to claim 57, wherein said clients in said group and/or further parties in said system interchange leaf hash values with one or more other group of clients and thereby compute non-leaf hash values and said root hash value.

59.   (Currently Amended) The method according to claim 58, wherein one or more clients of each group of clients accomplish the functionality of ~~said server of any one of claims 25 to 29.~~distributing the root hash value.

60.   (Currently Amended) The method according to claim 58 ~~or 59~~, wherein said server is a client in each group of clients.

61.   (Currently Amended) A computer-readable medium comprising instructions for controlling a client device in a client- server system, said instructions causing said client device to participate in a method according to any one of claims 32 to 60.

62.   (Currently Amended) A computer-readable medium comprising instructions for controlling a server in a client- server system, said instructions causing said server to participate in a method according to any one of claims 32 to 60.